

Surprises, Bugs, and Deep Dives: Lessons from a Python Upgrade

PyWeb, July 2025

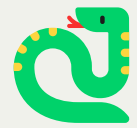
bit.ly/py312-upgrade

Yonatan Bitton
@bityob



whoami

- Developer on the Core team at **Fortinet**
- We provide cybersecurity protection for emails and more
- Passionate about **Python**, enjoy solving tough **challenges** and understanding things **inside out**



Python 3.9 → 3.12: The Upgrade Journey

Agenda / Upgrade Trail Map

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?
- Stage 1: Build Python 3.12

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?
- Stage 1: Build Python 3.12
- Stage 2: Installing & Resolving Package Issues

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?
- Stage 1: Build Python 3.12
- Stage 2: Installing & Resolving Package Issues
- Stage 3: Running & Fixing

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?
- Stage 1: Build Python 3.12
- Stage 2: Installing & Resolving Package Issues
- Stage 3: Running & Fixing
- Stage 4: Running 4k Tests — Reality Check

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?
- Stage 1: Build Python 3.12
- Stage 2: Installing & Resolving Package Issues
- Stage 3: Running & Fixing
- Stage 4: Running 4k Tests — Reality Check
- Stage 5: QA + ~500 automation end-to-end flows

Agenda / Upgrade Trail Map

- Stage 0: Why upgrade? Why 3.12? Strategy?
- Stage 1: Build Python 3.12
- Stage 2: Installing & Resolving Package Issues
- Stage 3: Running & Fixing
- Stage 4: Running 4k Tests — Reality Check
- Stage 5: QA + ~500 automation end-to-end flows
- Stage 6: Production rollout & real-world bugs

Day 0

Stage 0: Why Upgrade? Why 3.12?

Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure

Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure
- 3.12 is fast, stable, mature

Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure
- 3.12 is fast, stable, mature
- 3.13 is too new — risky deps

Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure
- 3.12 is fast, stable, mature
- 3.13 is too new — risky deps
- Large-scale pipeline — emails, files, links → scan → forward

Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure
- 3.12 is fast, stable, mature
- 3.13 is too new — risky deps
- Large-scale pipeline — emails, files, links → scan → forward
- Thousands of pods, millions of scans hourly

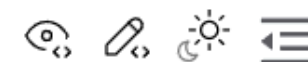
Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure
- 3.12 is fast, stable, mature
- 3.13 is too new — risky deps
- Large-scale pipeline — emails, files, links → scan → forward
- Thousands of pods, millions of scans hourly
- Strategy: 2 full local envs (3.9 & 3.12), side by side

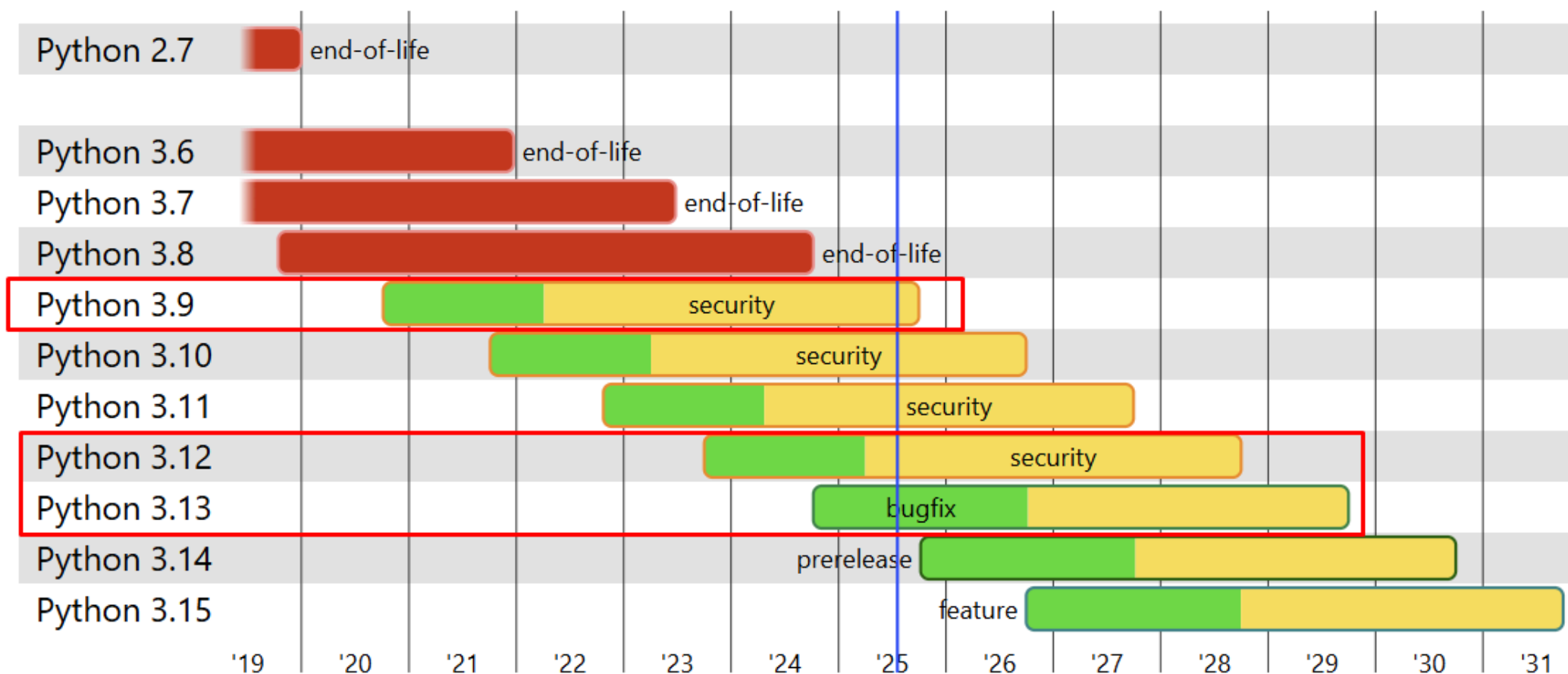
Stage 0: Why Upgrade? Why 3.12?

- 3.9 is EOL — security & ecosystem pressure
- 3.12 is fast, stable, mature
- 3.13 is too new — risky deps
- Large-scale pipeline — emails, files, links → scan → forward
- Thousands of pods, millions of scans hourly
- Strategy: 2 full local envs (3.9 & 3.12), side by side
- Compare outputs, debug diffs, document all

Status of Python versions



The `main` branch is currently the future Python 3.15, and is the only branch that accepts new features. The latest release for each Python version can be found on the [download page](#).



Day 1

Stage 1: Build Python 3.12

Stage 1: Build Python 3.12

- Need OpenSSL 1.1.1 — not v3

Stage 1: Build Python 3.12

- Need OpenSSL 1.1.1 — not v3
- OpenSSL 3 blocks legacy SMTP servers

Stage 1: Build Python 3.12

- Need OpenSSL 1.1.1 — not v3
- OpenSSL 3 blocks legacy SMTP servers
- Recompiled Python 3.12 with OpenSSL 1.1.1

Stage 1: Build Python 3.12

- Need OpenSSL 1.1.1 — not v3
- OpenSSL 3 blocks legacy SMTP servers
- Recompiled Python 3.12 with OpenSSL 1.1.1
- Packed to ZIP → used in Docker build.

Stage 1: Build Details

```
1 # Downloading required python source
2 wget https://www.python.org/ftp/python/3.12.9/Python-3.12.9.tgz
3 tar -xvf Python-3.12.9.tgz
4 cd Python-3.12.9
5
6 # Building python with openssl and pip to dedicated path
7 ./configure -C --with-ensurepip=install --with-openssl=/usr/local/ssl --prefix=/Python-3.12.9
8 make clean
9 make -s -j4
10 ./python -m ensurepip --default-pip
11 make install
12
13 # Saving the compiled python to zip file for later use
14 cd ..
15 tar -cvzf Python-3.12.9-openssl-1.1.1.built.tgz ./Python-3.12.9/compiled
```


Stage 1: Build Details

```
1 # Downloading required python source
2 wget https://www.python.org/ftp/python/3.12.9/Python-3.12.9.tgz
3 tar -xvf Python-3.12.9.tgz
4 cd Python-3.12.9
5
6 # Building python with openssl and pip to dedicated path
7 ./configure -C --with-ensurepip=install --with-openssl=/usr/local/ssl --prefix=/Python-3.12.9
8 make clean
9 make -s -j4
10 ./python -m ensurepip --default-pip
11 make install
12
13 # Saving the compiled python to zip file for later use
14 cd ..
15 tar -cvzf Python-3.12.9-openssl-1.1.1.built.tgz ./Python-3.12.9/compiled
```

Stage 1: Build Details

```
1 # Downloading required python source
2 wget https://www.python.org/ftp/python/3.12.9/Python-3.12.9.tgz
3 tar -xvf Python-3.12.9.tgz
4 cd Python-3.12.9
5
6 # Building python with openssl and pip to dedicated path
7 ./configure -C --with-ensurepip=install --with-openssl=/usr/local/ssl --prefix=/Python-3.12.9
8 make clean
9 make -s -j4
10 ./python -m ensurepip --default-pip
11 make install
12
13 # Saving the compiled python to zip file for later use
14 cd ..
15 tar -cvzf Python-3.12.9-openssl-1.1.1.built.tgz ./Python-3.12.9/compiled
```

Stage 1: Build Usage

```
RUN wget https://fake.io/Python-3.12.9-openssl-1.1.1.built.tgz -O Python.tgz &&\  
    tar -xvf Python.tgz && \  
    cp ./Python-3.12.9/compiled/* /usr/local/ -r  
RUN python3 -c "import ssl; assert '1.1.1' in ssl.OPENSSL_VERSION"
```

Stage 2: Installing & Resolving Package Issues

Stage 2: Installing & Resolving Package Issues

- Now Python 3.12 is ready to use

Stage 2: Installing & Resolving Package Issues

- Now Python 3.12 is ready to use
- Install packages using `uv sync` (similar to `pip install`)

Stage 2: Installing & Resolving Package Issues

- Now Python 3.12 is ready to use
- Install packages using `uv sync` (similar to `pip install`)
- Expect install errors due to 3.12 incompatibilities

Stage 2: Installing & Resolving Package Issues

- Now Python 3.12 is ready to use
- Install packages using `uv sync` (similar to `pip install`)
- Expect install errors due to 3.12 incompatibilities
- Upgrade to latest version if possible

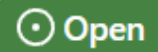
Stage 2: Installing & Resolving Package Issues

- Now Python 3.12 is ready to use
- Install packages using `uv sync` (similar to `pip install`)
- Expect install errors due to 3.12 incompatibilities
- Upgrade to latest version if possible
- Otherwise, find first version supporting Python 3.12

cchardet example

longintrepr.h: No such file or directory

```
54.71      -DNDEBUG -g -O3 -Wall -fPIC -fPIC -Isrc/ext/uchardet/src
54.71      -I/root/.cache/uv/builds-v0/.tmpgzo3a9/include
54.71      -I/root/.local/share/uv/python/cpython-3.12.9-linux-x86_64-gnu/include/python
54.71      -c src/cchardet/_cchardet.cpp -o
54.71      build/temp.linux-x86_64-cpython-312/src/cchardet/_cchardet.o
54.71
54.71      [stderr]
54.71      src/cchardet/_cchardet.cpp:196:12: fatal error: longintrepr.h: No such
54.71      file or directory
54.71      196 |     #include "longintrepr.h"
54.71          |           ^~~~~~
54.71      compilation terminated.
54.71      error: command '/usr/bin/c++' failed with exit code 1
54.71
54.71      hint: This error likely indicates that you need to install a library
54.71      that provides "longintrepr.h" for `cchardet@2.1.7`
54.71      help: `cchardet` (v2.1.7) was included because `mantis` (v1.0.0) depends
54.71      on `cchardet`
54.71      -----
```



Failed to Build on Python 3.11.0 #81

What is the problem?

Failed to build.

Expected behavior

Supposed to build.

Actual behavior

fatal error: longintrepr.h: No such file or directory

Steps to reproduce the behavior

Just update to Python 3.11 and attempt to build code results in error.



👍 20



- The non-limited API files `cellobject.h`, `classobject.h`, `code.h`, `context.h`, `funcobject.h`, `genobject.h` and `longintrepr.h` have been moved to the `Include/cpython` directory. Moreover, the `eval.h` header file was removed. These files must not be included directly, as they are already included in `Python.h`: [Include Files](#). If they have been included directly, consider including `Python.h` instead. (Contributed by Victor Stinner in [bpo-35134](#).)

Package didn't update in last years

cchardet 2.1.7

```
pip install cchardet
```



[Latest version](#)

Released: Oct 27, 2020

cChardet is high speed universal character encoding detector.

But there is a pre-release version

Release history

[Release notifications](#) | [RSS feed](#) 

2.2.0a2 **PRE-RELEASE**

Jun 6, 2024

THIS VERSION

2.1.7

Oct 27, 2020

There is also a fork option

cChardet

NOTICE: This is a fork of the original project at <https://github.com/PyYoshi/cChardet> since the original project is no longer maintained.

To install:

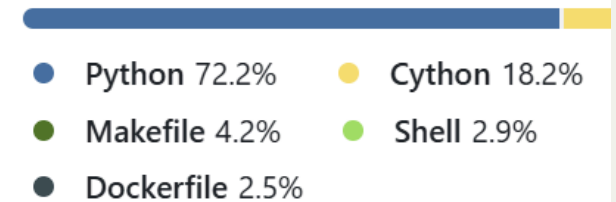
```
pip install faust-cchardet
```



 6 forks

 13 years old

 v2.1.19 Latest
on Aug 9, 2023



Report repository

Untitled query Run Save Download Share

```
1 SELECT project, file.version, count(*) as count
2 FROM `bigquery-public-data.pypi.file_downloads`
3 WHERE timestamp >= TIMESTAMP_SUB(CURRENT_TIMESTAMP(), INTERVAL 30 DAY)
4 | AND project = 'cchardet'
5 GROUP BY project, file.version
6 order by count desc
```

Query results

Job information	Results	Chart	JSON	Execution details	Execution g
Row	project	version	count		
1	cchardet	2.1.7	606845		
2	cchardet	2.1.5	27459		
3	cchardet	2.2.0a2	25740		
4	cchardet	2.1.6	9942		
5	cchardet	2.1.4	6384		

Query results

Job information

Results

Chart

JSON

Execution details

Execu

Row	project ▾	version ▾	count ▾
1	faust-cchardet	2.1.19	435618
2	faust-cchardet	2.1.18	25573
3	faust-cchardet	2.1.13	2073
4	faust-cchardet	2.1.16	1796
5	faust-cchardet	2.1.15	994
6	faust-cchardet	2.1.14	796
7	faust-cchardet	2.1.9rc5	762
8	faust-cchardet	2.1.12	719
9	faust-cchardet	2.1.12rc0	716
10	faust-cchardet	2.1.9	710
11	faust-cchardet	2.1.11	546

Although a popular fork exists, I chose to stick with the original `cchardet` after verifying that the pre-release only included safe recompilation changes — switching packages entirely felt riskier.

numpy example

```
> [build 4/5] RUN --mount=type=cache,target=/root/.cache/uv uv --directory /opt/app sync:
1.487 Downloading cpython-3.12.9-linux-x86_64-gnu (20.3MiB)
8.039 Downloaded cpython-3.12.9-linux-x86_64-gnu
8.502 Using CPython 3.12.9
8.506 Creating virtual environment at: venv
9.356 Building numpy==1.22.4
9.414 Building pycountry==17.5.14
9.417 Building scikit-learn==1.1.1
9.418 Building pylzma==0.5.0
9.421 Building capstone==3.0.4
9.422 Building pcre2==0.4.0
9.424 Building regex==2021.3.17
9.425 Building brotli==1.0.9
9.428 Building snowflake-conector-python==3.0.1
9.429 Building psycopg2==2.9.3
9.431 Building xmlsec==1.3.13
9.658 Downloading xgboost (184.0MiB)
9.730 Downloading opencv-python (59.0MiB)
9.732 Downloading pymupdfb (15.2MiB)
9.734 Downloading scipy (36.4MiB)
9.738 Downloading pyqt5-qt5 (58.3MiB)
9.898 Downloading ddtrace (6.0MiB)
18.79 x Failed to build 'numpy==1.22.4'
18.79 |→ The build backend returned an error
18.79 |→ Call to 'setuptools.build_meta:__legacy__.build_wheel' failed (exit
18.79 status: 1)
```

Python 3.12 removed distutils

```
18.79      hint: 'distutils' was removed from the standard library in Python 3.12.  
18.79      Consider adding a constraint (like 'numpy >1.22.4') to avoid building a  
18.79      version of 'numpy' that depends on 'distutils'.  
18.79      help: 'numpy' (v1.22.4) was included because 'mantis' (v1.0.0) depends on  
18.79      'numpy'
```

Dockerfile:99

```
-----  
 98 |      COPY pyproject.toml uv.lock $APP_DIR  
 99 | >>> RUN --mount=type=cache,target=/root/.cache/uv \  
100 | >>>      uv --directory $APP_DIR sync  
101 |      RUN cd $APP_DIR &&\
```

We can't upgrade numpy because an internal dependency is pinned to this specific version

```
> $ uv lock
x No solution found when resolving dependencies for split:
└─▶ Because some-internal-package==1.0.67 depends on numpy==1.22.4 and your project depends
    on numpy>1.22.4,
    we can conclude that your project and some-internal-package==1.0.67 are incompatible.
    And because your project depends on some-internal-package==1.0.67,
    we can conclude that your project's requirements are unsatisfiable.
```

[Code](#)[Blame](#)

35 lines (32 loc) · 754 Bytes

```
1  from setuptools import setup
2
3  ✓ REQUIREMENTS = [
4      "networkx",
5      "numpy==1.22.4",
6      "oletools==0.60",
7      "pyparsing==2.3.0",
8      "xgboost==1.6.1",
9      "scikit-learn==1.1.1",
10 ]
11
```


uv to the rescue!

Dependency overrides

Dependency overrides allow bypassing unsuccessful or undesirable resolutions by overriding a package's declared dependencies. Overrides are a useful last resort for cases in which you *know* that a dependency is compatible with a certain version of a package, despite the metadata indicating otherwise.

Unblocked upgrade with `override-metadata`; sent PR to fix deps.

```
[[tool.uv.dependency-metadata]]
name = "some-internal-package"
version = "1.0.67"
requires-dist = [
    #...
    "numpy>1.22.4",
    #...
]
```

Day 3

Stage 3: Running & Fixing

Stage 3: Running & Fixing

- First full run under Python 3.12 exposed multiple issues

Stage 3: Running & Fixing

- First full run under Python 3.12 exposed multiple issues
- This stage is mostly trial-and-error: issues only revealed themselves at runtime, requiring quick investigation and targeted fixes

Stage 3: Running & Fixing

- First full run under Python 3.12 exposed multiple issues
- This stage is mostly trial-and-error: issues only revealed themselves at runtime, requiring quick investigation and targeted fixes
- Common problems: missing imports, deprecated APIs, incompatible packages

Stage 3: Running & Fixing

- First full run under Python 3.12 exposed multiple issues
- This stage is mostly trial-and-error: issues only revealed themselves at runtime, requiring quick investigation and targeted fixes
- Common problems: missing imports, deprecated APIs, incompatible packages
- After several iterations:  system booted and app ran locally

pinecone example

```
File "/opt/app/venv/lib/python3.12/site-packages/django/apps/__init__.py", line 27, in setup
    apps.populate(settings.INSTALLED_APPS)
File "/opt/app/venv/lib/python3.12/site-packages/django/apps/registry.py", line 116, in populate
    app_config.import_models()
File "/opt/app/venv/lib/python3.12/site-packages/django/apps/config.py", line 269, in import_models
    self.models_module = import_module(models_module_name)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "/usr/local/lib/python3.12/importlib/__init__.py", line 90, in import_module
    return _bootstrap._gcd_import(name[level:], package, level)
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "<frozen importlib._bootstrap>", line 1387, in _gcd_import
File "<frozen importlib._bootstrap>", line 1360, in _find_and_load
File "<frozen importlib._bootstrap>", line 1331, in _find_and_load_unlocked
File "<frozen importlib._bootstrap>", line 935, in _load_unlocked
File "<frozen importlib._bootstrap_external>", line 999, in exec_module
File "<frozen importlib._bootstrap>", line 488, in _call_with_frames_removed
File "/opt/app/.../signatures/models.py", line 40, in <module>
    from ...core.pinecone_client import (
File "/opt/app/.../core/pinecone_client.py", line 1, in <module>
    import pinecone
File "/opt/app/venv/lib/python3.12/site-packages/pinecone/__init__.py", line 11, in <module>
    from .core.grpc.index_grpc import *
File "/opt/app/venv/lib/python3.12/site-packages/pinecone/core/grpc/index_grpc.py", line 23, in <module>
    from pinecone.core.grpc.protos.vector_service_pb2 import Vector as GRPCVector, \
File "/opt/app/venv/lib/python3.12/site-packages/pinecone/core/grpc/protos/vector_service_pb2.py", line 19, in <module>
    from google.api import annotations_pb2 as google_dot_api_dot_annotations__pb2
File "/opt/app/venv/lib/python3.12/site-packages/google/api/annotations_pb2.py", line 31, in <module>
    from google.api import http_pb2 as google_dot_api_dot_http__pb2
File "/opt/app/venv/lib/python3.12/site-packages/google/api/http_pb2.py", line 36, in <module>
    _HTTP = DESCRIPTOR.message_types_by_name["Http"]
            ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'message_types_by_name'
```

```
--  
name = "pinecone-client"  
version = "2.2.1"  
source = { registry = "https://pypi.org/simple" }  
dependencies = [  
  { name = "dnspython" },  
  { name = "loguru" },  
  { name = "numpy" },  
  { name = "python-dateutil" },  
  { name = "pyyaml" },  
  { name = "requests" },  
  { name = "tqdm" },  
  { name = "typing-extensions" },  
  { name = "urllib3" },  
]  
sdist = { url = "https://files.pythonhosted.org/packages"
```

pinecone-client 6.0.0

```
pip install pinecone-client
```



[Latest version](#)

Released: Feb 21, 2025

Pinecone client (DEPRECATED)

Pinecone Python Client (DEPRECATED)

The official Pinecone python package has been renamed from `pinecone-client` to `pinecone`. To upgrade, please **remove** `pinecone-client` from your project dependencies and then **add** the `pinecone` package to get the latest updates.

pinecone 7.3.0

```
pip install pinecone
```



Latest version

Released: Jun 27, 2025

Pinecone client and SDK

But, why we failed on `pinecone[grpc]` which we didn't install?

Looks good, no?

```
1  #
2  # Copyright (c) 2020-2021 Pinecone Systems Inc. All right reserved.
3  #
4  from pinecone.core.utils.constants import CLIENT_VERSION as __version__
5  from .config import *
6  from .exceptions import *
7  from .info import *
8  from .manage import *
9  from .index import *
10 try:
11      from .core.grpc.index_grpc import *
12 except ImportError:
13     pass # ignore for non-[grpc] installations
```



```

1  #
2  # Copyright (c) 2020-2021 Pinecone Systems Inc. All right reserved.
3  #
4  import logging
5  import numbers
6  from abc import ABC, abstractmethod
7  from functools import wraps
8  from importlib.util import find_spec
9  from typing import NamedTuple, Optional, Dict, Iterable, Union, List, Tuple, Any
10 from collections.abc import Mapping
11
12 import certifi
13 import grpc
14 from google.protobuf import json_format
15 from grpc._channel import _InactiveRpcError, _MultiThreadedRendezvous
16 from tqdm.autonotebook import tqdm
17 import json
18
19 from pinecone import FetchResponse, QueryResponse, ScoredVector, SingleQueryResults, DescribeIndexStatsResp
20 from pinecone.config import Config
21 from pinecone.core.client.model.namespace_summary import NamespaceSummary
22 from pinecone.core.client.model.vector import Vector as _Vector
23 from pinecone.core.grpc.protos.vector_service_pb2 import Vector as GRPCVector, \
24     QueryVector as GRPCQueryVector, UpsertRequest, UpsertResponse, DeleteRequest, QueryRequest, \
25     FetchRequest, UpdateRequest, DescribeIndexStatsRequest, DeleteResponse, UpdateResponse, \

```

Pinecone expected a missing `grpc` to raise `ImportError`, but it was installed via an unrelated package with the wrong version — leading to `AttributeError` instead

Day 5

Stage 4: Running 4k Tests — Reality Check

Stage 4: Running 4k Tests — Reality Check

- Ran full test suite on GitHub & locally

Stage 4: Running 4k Tests — Reality Check

- Ran full test suite on GitHub & locally
- Most tests passed, but some (~100) failed

Stage 4: Running 4k Tests — Reality Check

- Ran full test suite on GitHub & locally
 - Most tests passed, but some (~100) failed
 - Next: digging into a few tricky and interesting issues found
-

patool example

Tests are failing with an `IndexError` for some reason...

```

E ...Full output truncated (3 lines hidden), use '-vv' to show
short test summary info
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, leaf -> child -> root, leaf started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, root -> leaf -> child, leaf started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, leaf -> child -> root, leaf started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, root -> leaf -> child, leaf started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, leaf -> child -> root, leaf not started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, root -> leaf -> child, leaf not started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, leaf -> child -> root, leaf not started] - IndexError: list index out of range
FAILED ../tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, root -> leaf -> child, leaf not started] - IndexError: list index out of range

```

Let's debug it with pytest --trace

```
$BIN_PATH/pytest ./tests --lf -k test_mark_as_failed_passes_mal_verdict_to_root --trace
```

```
85     def _extract_to_outdir(self):
86         try:
87 ->         result = patoolib.extract_archive(self.file_path, outdir=self.output_dir, verbosity=-1, interactive=False)
88             self._handle_extract_result(result)
89             return True
90         except RuntimeError as e:
91             self.logger.warning('Archive is encrypted and currently not supported. msg={}'.format(e))
92         except patoolib.util.PatoolError as e:
93             self.logger.warning('Archive read error. msg={}. Attempting to extract files with additional software:'.
94                                 format(e))
95             if isinstance(e, patoolib.util.PatoolCommandError):
96                 return self._try_handle_extract_errors(e.command_result)
97             return False
PatoolError: unknown archive format for file '/tmp/tmpxhyejscopy'
(Pdb++)
```

```
PatoolError: unknown archive format for file `/tmp/tmpxhyejsxy'
```

Verified in both environments that the patool version **did not** change

```
97         return False
PatoolError: unknown archive format for file `/tmp/tmpxhyejsty'
(Pdb++) patoolib
<module 'patoolib' from '/opt/app/venv/lib/python3.12/site-packages/patoolib/__init__.py'>
(Pdb++) patoolib.__version__
'1.13.11'
(Pdb++) import hashlib
(Pdb++) from pathlib import Path
(Pdb++) hashlib.sha256(Path(self.file_path).read_bytes()).hexdigest()
'44bf95cc3e01c398960e6275f94ba5aa30700fe0a2cfab8b77c93cc799875a90'
(Pdb++) █
```

```
(Pdb++) w
[0] /usr/local/lib/python3.9/bdb.py(623)runcall()
-> res = func(*args, **kwargs)
[1] /opt/app/ /tests/test_core/test_mark_as_failed.py(71)test_mark_as_failed_passes_mal_verdict_to_root()
-> scan_tree = _create_scan_tree()
[2] /opt/app/ /tests/test_core/test_mark_as_failed.py(38)_create_scan_tree()
-> child_scan = _extract(scan)
[3] /opt/app/ /tests/test_core/test_mark_as_failed.py(25)_extract()
-> children = extract_scan(scan=scan, logger=scan.logger, create_children=True,
[4] /opt/app/ /unpackers/extractors/__init__.py(230)extract_scan()
-> extractor_children = extractor.extract() or []
[5] /opt/app/ /unpackers/extractors/archive.py(75)extract()
-> successfully_extracted = self._extract_to_outdir()
[6] > /opt/app/ /unpackers/extractors/archive.py(88)_extract_to_outdir()
-> self._handle_extract_result(result)
(Pdb++) patoolib.__version__
'1.13.11'
(Pdb++) self.file_path
'/tmp/tmpyyd70u13'
(Pdb++) import hashlib
(Pdb++) #hashlib.sha256(
*** SyntaxError: unexpected EOF while parsing
(Pdb++) from pathlib import Path
(Pdb++) hashlib.sha256(Path(self.file_path).read_bytes()).hexdigest()
'44bf95cc3e01c398960e6275f94ba5aa30700fe0a2cfab8b77c93cc799875a90'
(Pdb++)
```

Confirmed that the input file used was **exactly the same**

Let's go deeper...

```
patoolib.get_archive_format(self.file_path)
```

```
(Pdb++) patoolib.util.check_existing_filename(self.file_path)
(Pdb++) patoolib.get_archive_format(self.file_path)
*** patoolib.util.PatoolError: unknown archive format for file '/tmp/tmpxhyejsty'
Traceback (most recent call last):
  File "/opt/app/venv/lib/python3.12/site-packages/patoolib/__init__.py", line 316, in get_archive_format
    raise util.PatoolError("unknown archive format for file '%s'" % filename)
(Pdb++)
```

```
(Pdb++) patoolib.get_archive_format(self.file_path)
('zip', None)
(Pdb++)
```



```
11 
12 def get_archive_format(filename):
13     """Detect filename archive format and optional compression."""
14     mime, compression = util.guess_mime(filename)
15     if not (mime or compression):
16         raise util.PatoolError("unknown archive format for file '%s'" % filename)
17     if mime in ArchiveMimetypes:
18         format = ArchiveMimetypes[mime]
19     else:
20         raise util.PatoolError("unknown archive format for file '%s' (mime-type is '%s')" % (fi
21     if format == compression:
22         # file cannot be in same format compressed
23         compression = None
24     return format, compression
```

```
patoolib.util.guess_mime(self.file_path)
```

```
(Pdb++) patoolib.util.guess_mime(self.file_path)  
(None, None)
```

```
(Pdb++) patoolib.util.guess_mime(self.file_path)  
(('application/zip', None))  
(Pdb++)
```

```
241 @memoized
242 def guess_mime (filename):
243     """Guess the MIME type of given filename using file(1) and if that
244 fails by looking at the filename extension with the Python mimetypes
245 module.
246
247 The result of this function is cached.
248 """
249     mime, encoding = guess_mime_file(filename)
250     if mime is None:
251         mime, encoding = guess_mime_mimedb(filename)
252     assert mime is not None or encoding is None
253     return mime, encoding
```

```
django@dd8aee805609:/opt/app$ which file
django@dd8aee805609:/opt/app$ ls -la /usr/bin/file
ls: cannot access '/usr/bin/file': No such file or directory
django@dd8aee805609:/opt/app$
```

```
django@PP109-YonatanB:/opt/app$ which file
/usr/bin/file
```

```
django@PP109-YonatanB:/opt/app$ file --version
file-5.41
```

We don't install `file` explicitly, so where it come from?

Checking for **reverse** dependency on **file**

```
$ apt-cache rdepends --installed file
file
Reverse Depends:
  libmagic1
  mailcap
```

```
django@PP109-YonatanB:/opt/app$ apt-cache rdepends --installed file
file
```

```
Reverse Depends:
```

```
libmagic1
```

```
mailcap
```

```
django@PP109-YonatanB:/opt/app$ apt-cache rdepends --installed libmagic1
```

```
libmagic1
```

```
Reverse Depends:
```

```
libmagic-mgc
```

```
libmagic-mgc
```

```
file
```

```
django@PP109-YonatanB:/opt/app$ apt-cache rdepends --installed libmagic-mgc
```

```
libmagic-mgc
```

```
Reverse Depends:
```

```
libmagic1
```

```
django@PP109-YonatanB:/opt/app$ apt-cache rdepends --installed mailcap
```

```
mailcap
```

```
Reverse Depends:
```

```
mime-support
```

```
django@PP109-YonatanB:/opt/app$ apt-cache rdepends --installed mime-support
```

```
mime-support
```

```
Reverse Depends:
```

```
mailcap
```

```
python3.9
```

```
python3.10
```

```
media-types
```

```
media-types
```

```
mailcap
```

```
libpython3.10-stdlib
```

```
libpython3.9-stdlib
```

Since no python3.9, must install mime-support explicitly

```
root@dd8aee805609:/opt/app# apt install mime-support
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  file mailcap
The following NEW packages will be installed:
  file mailcap mime-support
0 upgraded, 3 newly installed, 0 to remove and 13 not upgraded.
Need to get 49.0 kB of archives.
After this operation, 194 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```


And indeed, after the fix, the tests are passing successfully

```
/usr/bin/file
django@ddd8aee805609:/opt/app$ $BIN_PATH/pytest /tests --lf -k test_mark_as_failed_passes_mal_verdict_to_root
/opt/app/venv/lib/python3.12/site-packages/sqlalchemy/orm/query.py:196: SyntaxWarning: "is not" with 'tuple' literal. Did you mean "!="?
    if entities is not ():
===== test session starts =====
platform linux -- Python 3.12.9, pytest-7.2.2, pluggy-0.13.1 -- /opt/app/venv/bin/python3
cachedir: /tmp/.cache
django: settings: config.settings.local (from env)
metadata: {'Python': '3.12.9', 'Platform': 'Linux-5.15.153.1-microsoft-standard-WSL2-x86_64-with-glibc2.35', 'Packages': {'pytest': '7.2.2', 'pluggy': '0.13.0', 'ddtrace': '3.1.0', 'xdist': '3.2.1', 'django': '3.10.0', 'allure-pytest': '2.6.0', 'cov': '2.4.0', 'rerunfailures': '14.0', 'mock': '2.0.0', 'timeout': '1.3.4', 'factoryboy': '2.5.1', 'requests-mock': '1.10.0', 'flaky': '3.8.1', 'env': '0.6.2', 'typeguard': '4.3.0'}, 'AWS_REGION': 'us-east-1'}
rootdir: /opt/app, testpaths: /tests, configfile: pytest.ini
plugins: anyio-4.7.0, ddtrace-3.1.0, xdist-3.2.1, django-3.10.0, allure-pytest-2.6.0, cov-2.4.0, rerunfailures-14.0, mock-2.0.0, timeout-1.3.4, html-1.22.1, requests-mock-1.10.0, flaky-3.8.1, env-0.6.2, typeguard-4.3.0
collected 474 items / 466 deselected / 8 selected
run-last-failure: rerun previous 8 failures (skipped 218 files)

./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, leaf -> child -> root, leaf started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, root -> leaf -> child, leaf started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, leaf -> child -> root, leaf started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, root -> leaf -> child, leaf started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, leaf -> child -> root, leaf not started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[leaf mal, root -> leaf -> child, leaf not started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, leaf -> child -> root, leaf not started] PASSED
./tests/test_core/test_mark_as_failed.py::test_mark_as_failed_passes_mal_verdict_to_root[child mal, root -> leaf -> child, leaf not started] PASSED

===== 8 passed, 466 deselected in 16.65s =====
```

URL parsing example

--000000000000006937e605c0507a5d

Content-Type: text/plain; charset="UTF-8"

http:\\google.com -

ftp://localhost/ -

http:/google.com

https:/google.com

ftp:/google.com

http:google.com

```
# python3.9
>>> find_urls_in_data("http:google.com")
[
    'http:///google.com'
]
```

```
# python3.12
>>> find_urls_in_data("http:google.com")
[
    'http:google.com'
]
```

```
def find_urls_in_data(data)
    results = []
    ....
    for url in found_urls:
        # Until now, the same result on both versions
        results.append(
            # Meaning the problem is here...
            safe_url_decode(url)
        )
    return results
```

```
def safe_url_decode(data)
    ...
    furl_url = furl.furl(url).url
    ...
```

```
In [1]: import sys; sys.version
```

```
Out[1]: '3.9.18 (main, Sep 5 2023, 22:42:59) \n[GCC 11.4.0]'
```

```
In [2]: import furl
```

```
In [3]: furl.furl("http:google.com").url
```

```
Out[3]: 'http:///google.com'
```

```
In [4]: furl.__version__
```

```
Out[4]: '2.1.3'
```



```
In [1]: import sys; sys.version
```

```
Out[1]: '3.12.9 (main, Mar  5 2025, 02:25:27) [GCC 11.4.0]'
```

```
In [2]: import furl
```

```
In [3]: furl.furl("http:google.com").url
```

```
Out[3]: 'http:google.com'
```

```
In [4]: furl.__version__
```

```
Out[4]: '2.1.3'
```

```
1337     class furl(URLPathCompositionInterface, QueryCompositionInterface,
1822         def tostr(self, query_delimiter='&', query_quote_plus=True,
1823                 query_dont_quote=False):
1824             encoded_query = self.query.encode(
1825                 query_delimiter, query_quote_plus, query_dont_quote)
1826             url = urllib.parse.urlunsplit((
1827                 self.scheme or '', # Must be text type in Python 3.
1828                 self.netloc,
1829                 str(self.path),
1830                 encoded_query,
1831                 str(self.fragment),
1832             ))
```

```
In [7]: import sys; sys.version
```

```
Out[7]: '3.9.18 (main, Sep 5 2023, 22:42:59) \n[GCC 11.4.0]'
```

```
In [8]: import urllib
```

```
In [9]: print(urllib.parse.urlunsplit(("http", "", "google.com", "", "")))  
http:///google.com
```

```
In [5]: import sys; sys.version
```

```
Out[5]: '3.12.9 (main, Mar  5 2025, 02:25:27) [GCC 11.4.0]'
```

```
In [6]: import urllib
```

```
In [7]: print(urllib.parse.urlunsplit(("http", "", "google.com", "", "")))  
http:google.com
```

```
for i in {9..12}; \  
do uvx --python 3.$i ipython -c \  
  'import sys; print(sys.version); import urllib; urllib.parse.urlunsplit(("http", "", "google"  
done
```

3.9.18 (main, Feb 25 2024, 04:21:36)

[Clang 17.0.6]

Out[1]: 'http://google.com'

3.10.15 (main, Oct 16 2024, 04:37:23) [Clang 18.1.8]

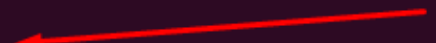
Out[1]: 'http://google.com'

3.11.10 (main, Oct 16 2024, 04:38:48) [Clang 18.1.8]

Out[1]: 'http://google.com'

3.12.10 (main, Apr 9 2025, 04:03:51) [Clang 20.1.0]

Out[1]: 'http:google.com'



```
for i in {1..9}; \  
do uvx --python 3.12.$i ipython -c \  
  'import sys; print(sys.version); import urllib; urllib.parse.urlunsplit(("http", "", "google"  
done
```

3.12.1 (main, Jan 8 2024, 05:57:25) [Clang 17.0.6]

Out[1]: 'http:///google.com'

3.12.2 (main, Feb 25 2024, 04:38:01) [Clang 17.0.6]

Out[1]: 'http:///google.com'

3.12.3 (main, Apr 15 2024, 18:25:56) [Clang 17.0.6]

Out[1]: 'http:///google.com'

3.12.4 (main, Jul 25 2024, 22:42:01) [Clang 18.1.8]

Out[1]: 'http:///google.com'

3.12.5 (main, Aug 14 2024, 05:08:31) [Clang 18.1.8]

Out[1]: 'http:///google.com'

3.12.6 (main, Sep 9 2024, 22:11:19) [Clang 18.1.8]

Out[1]: 'http:google.com' ←

18 years ago	 Merged revisions 55328-55341 v...		536	<code>def urlunsplit(components):</code>
15 years ago	 Merged revisions 82324 via svn...		537 538 539 540 541	<code>"""Combine the elements of a tuple as returned by urlsplit() into a complete URL as a string. The data argument can be any five-item iterable. This may result in a slightly different, but equivalent URL, if the URL that was parsed originally had unnecessary delimiters (for example, a ? with an empty query; the RFC states that these are equivalent)."""</code>
15 years ago	 Issue 9873: the URL parsing func...		542 543	<code>scheme, netloc, url, query, fragment, _coerce_result = (_coerce_args(*components))</code>
6 months ago	 [3.13] gh-85110 : Preserve relativ...		544	<code>if netloc:</code>
25 years ago	 Whitespace normalization. Top l...		545	<code>if url and url[:1] != '/': url = '/' + url</code>
6 months ago	 [3.13] gh-85110 : Preserve relativ...		546 547 548 549 550	<code>url = '/' + netloc + url elif url[:2] == '//': url = '/' + url elif scheme and scheme in uses_netloc and (not url or url[:1] == '/': url = '/' + url</code>
25 years ago	 Whitespace normalization. Top l...		551 552 553	<code>if scheme: url = scheme + ':' + url if query:</code>

urllib.parse.urlunsplit makes relative path to absolute (http:g -> http:///g) #85110

✓ Closed



openandclose (Open Close) **mannequin** opened on Jun 10, 2020 · edited by bedevere-app

Edits ▼

Mannequin



Assignees

No one assigned

Beautiful Soup Example

Test failed because email output is different

```
Everything Else ▾ ANSI ▾ PC
Return-Path: <automation@test.com>
MIME-Version: 1.0
Reply-To: Reply-to test2 <reply@to.com>
Sender: [REDACTED] <automation@test.com>
Message-ID: 1234
Date: Thu, 02 Jun 2022 06:12:35 +0000
Subject: whitelist-none
From: [REDACTED] <automation@test.com>
To: [REDACTED]
Content-Type: multipart/alternative; boundary="00000000000080e0c05d80dfb95"

--00000000000080e0c05d80dfb95
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: base64

[REDACTED]

--00000000000080e0c05d80dfb95
MIME-Version: 1.0
Content-Type: text/html; charset="UTF-8"
Content-Transfer-Encoding: base64

PGh0bWw+CiA8Ym9keT4KICA8ZG12IGRpcj0ibHRYIj4KICAgSEVMTE8KICA8L2Rpdj4KIDvvYm9k
eT4KPC9odG1sPgo=

--00000000000080e0c05d80dfb95--

Everything Else ▾ ANSI ▾ PC
Return-Path: <automation@test.com>
MIME-Version: 1.0
Reply-To: Reply-to test2 <reply@to.com>
Sender: [REDACTED] <automation@test.com>
Message-ID: 1234
Date: Thu, 02 Jun 2022 06:12:35 +0000
Subject: whitelist-none
From: [REDACTED] <automation@test.com>
To: [REDACTED]
Content-Type: multipart/alternative; boundary="00000000000080e0c05d80dfb95"

--00000000000080e0c05d80dfb95
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: base64

[REDACTED]

--00000000000080e0c05d80dfb95
MIME-Version: 1.0
Content-Type: text/html; charset="UTF-8"
Content-Transfer-Encoding: base64

PGh0bWw+PGJvZHK+PGRpdj0kaXI9Imx0ciI+SEVMTE/CoDvvZG12Pgo8L2JvZHK+PC9odG1sPg==

--00000000000080e0c05d80dfb95--
```

```
# python3.9
<html><body><div dir="ltr">HELLO </div>
</body></html>
```

```
# python 3.12
<html>
  <body>
    <div dir="ltr">
      HELLO
    </div>
  </body>
</html>
```

```
18 --0000000000000000000080e0c05d80dfb95
19 Content-Type: text/html; charset="UTF-8"
20 Content-Transfer-Encoding: quoted-printable
21
22 <div dir=3D"ltr">HELLO=C2=A0</div>
23
24 --0000000000000000000080e0c05d80dfb95--
25
```

Before set_payload, the payload looks the same

```
179  ->         new_msg.set_payload(  
180             payload,  
181             charset=EMLHandler.get_charset(old_msg),  
182         )  
183  
184         # Set headers to new_msg  
185         _copy_headers(old_msg, new_msg)  
186         return new_msg  
(Pdb++) payload  
<html><body><div dir="ltr">HELLO </div>  
</body></html>  
(Pdb++)
```

```
179         new_msg.set_payload(  
180             payload,  
181             charset=EMLHandler.get_charset(old_msg),  
182         )  
183  
184         # Set headers to new_msg  
185     ->     _copy_headers(old_msg, new_msg)  
186     return new_msg  
(Pdb++) str(new_msg)  
'MIME-Version: 1.0\r\nContent-Type: text/plain; charset="utf-8"\r\nContent-Transfer-  
eT4KPC9odG1sPgo=\r\n'  
(Pdb++) print(str(new_msg))  
MIME-Version: 1.0  
Content-Type: text/plain; charset="utf-8"  
Content-Transfer-Encoding: base64  
  
PGh0bWw+CiA8Ym9keT4KICA8ZGl2IGRpcj0ibHRyIj4KICAgSEVMTE8KICA8L2Rpdj4KIDwvYm9k  
eT4KPC9odG1sPgo=  
  
(Pdb++) █
```

But after set_payload, the payload looks different

```
b'<html>\n <body>\n  <div dir="ltr">\n    HELLO\n  </div>\n </body>\n</html>\n'  
(Pdb++) print(base64.b64decode(b"PGh0bWw+CiA8Ym9keT4KICA8ZGl2IGRpcj0ibHRyIj4KICAgSEVMT  
<html>  
  <body>  
    <div dir="ltr">  
      HELLO  
    </div>  
  </body>  
</html>
```

Let's try minimal repro in notebook...

Didn't work

```
msg = EmailMessage()
```

```
msg.set_payload("""<html><body><div dir="ltr">HELLO </div>  
</body></html>""", charset="utf-8")
```

```
print(str(msg))
```

MIME-Version: 1.0

Content-Type: text/plain; charset="utf-8"

Content-Transfer-Encoding: base64

PGh0bWw+PGJvZHK+PGRpdIBkaXI9Imx0ciI+SEVMTE/CoDwvZGl2Pgo8L2JvZHK+PC9odG1sPg==

```
import base64
```

```
print(base64.b64decode(b"PGh0bWw+PGJvZHK+PGRpdIBkaXI9Imx0ciI+SEVMTE/CoDwvZGl2Pgo8L2JvZHK+PC9odG1sPg==").decode())
```

```
<html><body><div dir="ltr">HELLO </div>  
</body></html>
```

There is an hidden byte in HELLO last char... trying with bytes

```
(Pdb++) msg3 = Message()
(Pdb++) msg3.set_payload(b'<html><body><div dir="ltr">HELLO\x02\xa0</div>\n</body></html>'.decode(),
(Pdb++) print(str(msg3))
MIME-Version: 1.0
Content-Type: text/plain; charset="utf-8"
Content-Transfer-Encoding: base64

PGh0bWw+PGJvZHK+PGRpdjBkaXI9Imx0ciI+SEVMTE/CoDwvZG12Pgo8L2JvZHK+PC9odG1sPg==
```

Let's use pickle to be sure it's the **real same object**

This is weird... this is **NOT** simple string

```
import pickle
pickle.dumps(payload)
```

```
b'\x00\x04\x95W\x04\x00\x00\x00\x00\x00\x00\x8c\x03bs4\x94\x8c\rBeautifulSoup\x94\x93\x94)\x81\x94}\x94(\x8c\x0felement_c1
lder._lxml\x94\x8c\x0fLXMLTreeBuilder\x94\x93\x94\x8c\x06is_xml\x94\x89\x8c\tknown_xml\x94\x89\x8c\x0b_namespaces\x94}\x94
amespace\x94s\x8c\nparse_only\x94N\x8c\x06markup\x94\x8c7<html><body><div dir="ltr">HELLO\xa0</div>\n</body>
</html>\x94\x8c\x11original_encoding\x94N\x8c\x16declared_html_encoding\x94N\x8c\x1fcontains_replacement_characters\x94\x8
8c\n[document]\x94\x8c\tnamespace\x94N\x8c\x06prefix\x94N\x8c\x05attrs\x94}\x94\x8c\x08contents\x94]\x94\x8c\x06parent\x94
nt\x94N\x8c\x0cnext_sibling\x94N\x8c\x10previous_sibling\x94N\x8c\x06hidden\x94K\x01\x8c\x14can_be_empty_element\x94\x89\
94]\x94(\x8c\x05class\x94\x8c\taccesskey\x94\x8c\x08dropzone\x94e\x8c\x01a\x94]\x94(\x8c\x03rel\x94\x8c\x03rev\x94e\x8c\x0
eaders\x94a\x8c\x02th\x94]\x94h6a\x8c\x04form\x94]\x94\x8c\x0eaccept-
charset\x94a\x8c\x06object\x94]\x94\x8c\x07archive\x94a\x8c\x04area\x94]\x94h0a\x8c\x04icon\x94]\x94\x8c\x05sizes\x94a\x8c
utput\x94]\x94\x8c\x03for\x94au\x8c\x18preserve_whitespace_tags\x94\x8f\x94(\x8c\x03pre\x94\x8c\x08textarea\x94\x90\x8c\x
\x94\x8c\x0fNavigableString\x94\x93\x94h0\x8c\x05CDATA\x94\x93\x94\x86\x94\x8c\x0ccurrent_data\x94]\x94\x8c\ncurrentTag\x9
n_tag_counter\x94\x8c\x0bcollections\x94\x8c\x07Counter\x94\x93\x94}\x94(\x8c\x04html\x94K\x00\x8c\x04body\x94K\x00\x8c\x0
espace_tag_stack\x94]\x94\x8c\x16string_container_stack\x94]\x94ub.'
```

What??

```
[10]: pickle.loads(b'\x80\x04\x95W\x04\x00\x00\x00\x00\x00\x00\x00\x8c\x03bs4\x94\x8c\rBeautifulSoup\x94
```

ModuleNotFoundError

Traceback (most recent call last)

Cell In[10], line 1

```
----> 1 pickle.loads( \x80\x04\x95 \x04\x00\x00\x00\x00\x00\x00\x00\x8c\x03 \x94\x8c\r
    \x94\x8c\x0f \x94\x93\x94\x8c\x06 \x94\x89\x8c\t \x94\x89\x8c\x0
c\x06 \x94\x8c \xc2\xa0 \n \x94\x8c\x1
\x8c\x0c \x94 \x02\x8c\x04 \x94\x8c\n \x94\x8c\t \x94 \x8c\x06
\x8c\x0c \x94 \x8c\x0c \x94 \x8c\x10 \x94 \x8c\x06
c\x05 \x94\x8c\t \x94\x8c\x08 \x94 \x8c\x01 \x94 \x94 \x8c\x03 \x94\x8c\
4 \x94 \x94\x8c\x0e \x94 \x8c\x06 \x94 \x94\x8c\x07 \x94 \x8c\x04
    \x94 \x94\x8c\x03 \x94 \x8c\x18 \x94\x8f\x94 \x8c\x03 \x94\x
94\x93\x94 \x8c\x05 \x94\x93\x94\x86\x94\x8c\x0c \x94 \x94\x8c\n \x94
4\x93\x94 \x94 \x8c\x04 \x94 \x00\x8c\x04 \x94 \x00\x8c\x03 \x94 \x00 \x85\x94 \x94\x
```

ModuleNotFoundError: No module named 'bs4'



```
(Pdb++) type(payload)  
<class 'bs4.BeautifulSoup'>
```

What's happening inside `set_payload`?

We call encode on the payload variable

```
335 * def set_payload(self, payload, charset=None):
336     """Set the payload to the given value.
337
338     Optional charset sets the message's default character set. See
339     set_charset() for details.
340     """
341     if hasattr(payload, 'encode'):
342         if charset is None:
343             self._payload = payload
344             return
345         if not isinstance(charset, Charset):
346             charset = Charset(charset)
347         payload = payload.encode(charset.output_charset, errors: 'surrogateescape')
348     if hasattr(payload, 'decode'):
```


Encoding bs4.BeautifulSoup vs encoding simple str

```
[95]: print(payload.encode("utf-8", 'surrogateescape').decode())
```

```
<html>
  <body>
    <div dir="ltr">
      HELLO
    </div>
  </body>
</html>
```

```
[89]: type(payload)
```

```
[89]: bs4.BeautifulSoup
```

```
[98]: print(str(payload).encode("utf-8", 'surrogateescape').decode())
```

```
<html><body><div dir="ltr">HELLO </div>
</body></html>
```

```
1662
1663 def encode(self, encoding=DEFAULT_OUTPUT_ENCODING,
1664             indent_level=None, formatter="minimal",
1665             errors="xmlcharrefreplace"):
1666     """Render a bytestring representation of this PageElement and its
1667     contents.
1668
1669     :param encoding: The destination encoding.
1670     :param indent_level: Each line of the rendering will be
1671         indented this many levels. (The formatter decides what a
1672         'level' means in terms of spaces or other characters
1673         output.) Used internally in recursive calls while
1674         pretty-printing.
1675     :param formatter: A Formatter object, or a string naming one of
1676         the standard formatters.
1677     :param errors: An error handling strategy such as
1678         'xmlcharrefreplace'. This value is passed along into
1679         encode() and its value should be one of the constants
1680         defined by Python.
1681     :return: A bytestring.
1682
1683     """
1684     # Turn the data structure into Unicode, then encode the
1685     # Unicode.
1686     u = self.decode(indent_level, encoding, formatter)
1687     return u.encode(encoding, errors)
```

18 years ago  Copying the email package back, ...

334

335

336

337

338

339

340

```
def set_payload(self, payload, charset=None):  
    """Set the payload to the given value.  
  
    Optional charset sets the message's default character set. See  
    set_charset() for details.  
    """
```

12 years ago  #20531: Apply the 3.3 version of... 

341

342

343

344

345

346

```
if hasattr(payload, 'encode'):  
    if charset is None:  
        self._payload = payload  
        return  
    if not isinstance(charset, Charset):  
        charset = Charset(charset)
```

last year  gh-76511: Fix email.Message.as_... 

347

```
payload = payload.encode(charset.output_charset, 'surrogateescape')
```

12 years ago  #20531: Apply the 3.3 version of... 

348

349

350

```
if hasattr(payload, 'decode'):  
    self._payload = payload.decode('ascii', 'surrogateescape')  
else:
```

The commit set errors value to surrogateescape

Merged

gh-76511: Fix email.Message.as_string() for non-ASCII message with ASCII charset #116125

0 / 4 files viewed

Ask Copilot

Review in codespace

Review changes

all commits

File filter

Conversations

Filter changed files

Lib

email

generator.py

message.py

test/test_email

test_email.py

Misc/NEWS.d/next/Library

2024-02-29-17-06-54.gh-issue...

Lib/email/generator.py

246

246

247

248

249

.....msg.set_payload(payload, charset)

.....msg.set_payload(msg._payload, charset)

.....payload = msg.get_payload()

.....self._munge_cte = (msg['content-transfer-encoding'],

.....msg['content-type'])

Lib/email/message.py

@@ -340,7 +340,7 @@ def set_payload(self, payload, charset=None):

340

340

341

341

342

342

343

343

344

345

346

.....return

.....if not isinstance(charset, Charset):

.....charset = Charset(charset)

.....payload = payload.encode(charset.output_charset)

.....payload = payload.encode(charset.output_charset, 'surrogateescape')

.....if hasattr(payload, 'decode'):

.....self._payload = payload.decode('ascii', 'surrogateescape')

.....else:

How this errors is related at all? why only on bs4 object?

```
[22]: print(payload.encode("utf-8", 'surrogateescape').decode())
```

```
<html>
<body>
  <div dir="ltr">
    HELLO
  </div>
</body>
</html>
```

```
[27]: print(payload.encode("utf-8").decode())
```

```
<html><body><div dir="ltr">HELLO </div>
</body></html>
```

```
[25]: print(str(payload).encode("utf-8", 'surrogateescape').decode())
```

```
<html><body><div dir="ltr">HELLO </div>
</body></html>
```

```
[26]: print(str(payload).encode("utf-8").decode())
```

```
<html><body><div dir="ltr">HELLO </div>
</body></html>
```

The surrogateescape string is going to indent_level variable...

```
1663     def encode(self, encoding=DEFAULT_OUTPUT_ENCODING,
1664                indent_level=None, formatter="minimal",
1665                errors="xmlcharrefreplace"):
1666         """Render a bytestring representation of this PageElement and its
1667         contents.
1668
1669         :param encoding: The destination encoding.
1670         :param indent_level: Each line of the rendering will be
1671             indented this many levels. (The formatter decides what a
1672             'level' means in terms of spaces or other characters
1673             output.) Used internally in recursive calls while
1674             pretty-printing.
1675         :param formatter: A Formatter object, or a string naming one of
1676             the standard formatters.
1677         :param errors: An error handling strategy such as
1678             'xmlcharrefreplace'. This value is passed along into
1679             encode() and its value should be one of the constants
1680             defined by Python.
1681         :return: A bytestring.
1682
1683         """
1684         # Turn the data structure into Unicode, then encode the
1685         # Unicode.
1686         u = self.decode(indent_level, encoding, formatter)
1687         return u.encode(encoding, errors)
```

No matter what string we use

```
[15]: print(payload.encode("utf-8", 'surrogateescape').decode())
```

```
<html>
<body>
  <div dir="ltr">
    HELLO
  </div>
</body>
</html>
```

```
[16]: print(payload.encode("utf-8", 'blablabla').decode())
```

```
<html>
<body>
  <div dir="ltr">
    HELLO
  </div>
</body>
</html>
```

It evaluates to True in a boolean context

```
777     def decode(self, pretty_print=False,  
778                 eventual_encoding=DEFAULT_OUTPUT_ENCODING,  
779                 formatter="minimal", iterator=None):  
780         """Returns a string or Unicode representation of the parse tree  
781            as an HTML or XML document.  
782  
783         :param pretty_print: If this is True, indentation will be used to  
784            make the document more readable.  
785         :param eventual_encoding: The encoding of the final document.  
786            If this is None, the document will be a Unicode string.  
787         """  
788         if self.is_xml:  
789             # Print the XML declaration  
790             encoding_part = ''  
791             if eventual_encoding in PYTHON_SPECIFIC_ENCODINGS:  
792                 # This is a special Python encoding; it can't actually  
793                 # go into an XML document because it means nothing  
794                 # outside of Python.  
795                 eventual_encoding = None  
796             if eventual_encoding != None:  
797                 encoding_part = ' encoding="%s"' % eventual_encoding  
798             prefix = '<?xml version="1.0"%s?>\n' % encoding_part  
799         else:  
800             prefix = ''  
801         if not pretty_print:  
802             indent_level = None  
803         else:  
804             -> indent_level = 0  
805         return prefix + super(BeautifulSoup, self).decode(  
806             indent_level, eventual_encoding, formatter, iterator)  
807  
(Pdb++) pretty_print  
'surrogateescape'  
(Pdb++)
```


Day 20

Stage 5: QA & End-to-End Testing

Stage 5: QA & End-to-End Testing

- Deployed to full-scale **testing environment**, mirroring production

Stage 5: QA & End-to-End Testing

- Deployed to full-scale **testing environment**, mirroring production
- **5 days** of QA: human exploratory testing + ~500 automated E2E tests

Stage 5: QA & End-to-End Testing

- Deployed to full-scale **testing environment**, mirroring production
 - **5 days** of QA: human exploratory testing + ~500 automated E2E tests
 - Monitored failures via Sentry, logs, and runtime alerts
-

SMTP starttls example

Issues >  [Python 3.7.0 - 3.7.1](#)

TypeError

 SMTP.starttls() got an unexpected keyword argument 'keyfile'

Escalating | [core.mail in open](#)

Stack Trace

Most Relevant

Full

TypeError

SMTP.starttls() got an unexpected keyword argument 'keyfile'

mechanism

logging

handled

true

.../core/mail.py in open at line 274

```
269
270         # TLS/SSL are mutually exclusive, so only attempt TLS over
271         # non-secure connections.
272         if not self.use_ssl and self.use_tls:
273             self.connection.ehlo(name=settings.NAT_DNS_NAME)
274             self.connection.starttls(keyfile=self.ssl_keyfile, certfile=self.ssl_certfile)
275             self.connection.ehlo(name=settings.NAT_DNS_NAME)
276         if self.username and self.password:
277             self.connection.login(self.username, self.password)
278         return True
```


SMTP.**starttls**(..., *context=None*)

Put the SMTP connection in TLS (Transport Layer Security) mode. All SMTP commands that follow will be encrypted. You should then call [ehlo\(\)](#) again.

If *keyfile* and *certfile* are provided, they are used to create an [ssl.SSLContext](#).

Optional *context* parameter is an [ssl.SSLContext](#) object; This is an alternative to using a keyfile and a certfile and if specified both *keyfile* and *certfile* should be `None`.

If there has been no previous `EHLO` or `HELO` command this session, this method tries ESMTP `EHLO` first.

Changed in version 3.12: The deprecated *keyfile* and *certfile* parameters have been removed.

```
1 if not self.use_ssl and self.use_tls:
2     self.connection.ehlo()
3     context = None
4     if self.ssl_certfile and self.ssl_keyfile:
5         context = ssl.create_default_context()
6         context.load_cert_chain(certfile=self.ssl_certfile, keyfile=self.ssl_keyfile)
7     self.connection.starttls(context=context)
8     self.connection.ehlo()
```

```
1 if not self.use_ssl and self.use_tls:
2     self.connection.ehlo()
3     context = None
4     if self.ssl_certfile and self.ssl_keyfile:
5         context = ssl.create_default_context()
6         context.load_cert_chain(certfile=self.ssl_certfile, keyfile=self.ssl_keyfile)
7     self.connection.starttls(context=context)
8     self.connection.ehlo()
```

SAML example

502 Bad Gateway

No exception in Sentry, only the worker died suddenly 🙌

Worker (pid:44) was sent code 136!

| (No error message)

🔗 -TESTING-191D1 | gunicorn.error

14s ago

2hr

Escalating

No logs either

```
testing-master- -web-internal-debug-67cf96c7c8-jnhg8 -web-internal-debug 10.10.119.27 - - [20/Apr/2025:11:59:15 +0000] "GET /metri
testing-master- -web-internal-559576fc7b-x54mm web internal 10.10.119.27 [20/Apr/2025:11:59:15 +0000] "GET /metrics HTTP/1.1"
testing-master- -web-69588b7b89-sfmk7 -web [2025-04-20 11:59:17 +0000] [9] [ERROR] Worker (pid:1083) was sent code 136!
testing-master- -web-69588b7b89-sfmk7 -web [2025-04-20 11:59:17 +0000] [1794] [INFO] Booting worker with pid: 1794
testing-master- -web-internal-559576fc7b-mjq7q -web-internal 10.10.119.27 - - [20/Apr/2025:11:59:17 +0000] "GET /metrics HTTP/1.1"
testing-master- -web-internal-559576fc7b-mjq7q -web-internal 10.10.119.27 - - [20/Apr/2025:11:59:17 +0000] "GET /metrics HTTP/1.1"
```

Let's try to reproduce the issue in shell

```
factory = APIRequestFactory()
user = User.objects.get(pk=12674)
payload = urlencode({'SAMLResponse': ''})
request = factory.post(
    '/api/saml_login/foobar/?acs',
    data=payload,
    content_type='application/x-www-form-urlencoded',
    HTTP_HOST="api.foo.bar.io",
    wsgi_url_scheme='https',
    SERVER_PORT='443',
)
force_authenticate(request, user=User.objects.get(pk=12674))
view = SamlAuthTokenViewSet.as_view({'post': 'post'})
view(request, uuid_id="f042fb42-4ac1-4e5a-bdb3-db286af56873")
```


Floating point exception?!

```
...: middleware.process_request(request)
...: request.session.save()
...: view = SamlAuthTokenViewSet.as_view({'post': 'post'})
...: view(request, uuid_id="+042+b42-4ac1-4e5a-bdb3-db286a+56873")
Floating point exception (core dumped)
django@testing-yobitton2-master-...-web-7d99b54665-qf5xv:/tmp$ ls -la
total 503908
drwxrwxrwt 1 root    root      23 Apr 20 17:42 .
drwxr-xr-x 1 root    root      86 Apr 20 15:04 ..
-rw----- 1 django  django 589205504 Apr 20 17:42 core.2874
-rwxrwxrwx 1 root    root        0 Apr 20 12:27 uv-6143060865219c88.lock
```

```
gdb $BIN_PATH/python core.2874
```

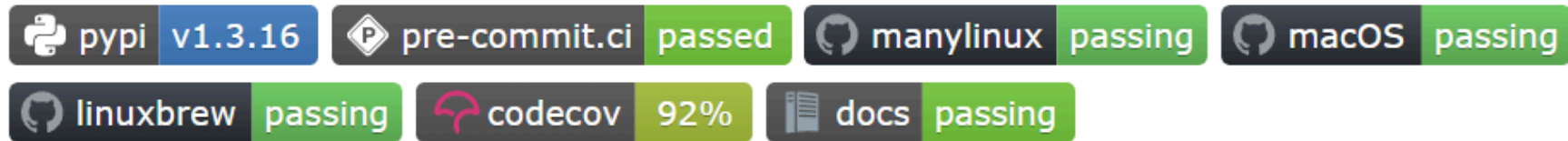
```
root@7631fff5c551:/opt/app# gdb $BIN_PATH/python core.2874
GNU gdb (Ubuntu 12.1-0ubuntu1~22.04.2) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
  <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from venv/bin/python...
[New LWP 2874]
[New LWP 2881]
[New LWP 2877]
[New LWP 2876]
[New LWP 2900]
[New LWP 2875]
```

Related to xmlsec package

```
[New LWP 2905]
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Core was generated by '/opt/app/venv/bin/python /opt/app/manage.py shell'.
Program terminated with signal SIGFPE, Arithmetic exception.
#0  0x00007fcdcb497ac1 in ?? () from /lib/x86_64-linux-gnu/libxml2.so.2
[Current thread is 1 (Thread 0x7fcdcb4042740 (LWP 2874))]
(gdb) bt
#0  0x00007fcdcb497ac1 in ?? () from /lib/x86_64-linux-gnu/libxml2.so.2
#1  0x00007fcdcb4a092ca0 in xmlHashLookup3 () from /lib/x86_64-linux-gnu/libxml2.so.2
#2  0x00007fcdcb4a09ffd7 in xmlGetID () from /lib/x86_64-linux-gnu/libxml2.so.2
#3  0x00007fcdcb43cea49 in xmlSecAddIDs () from /lib/x86_64-linux-gnu/libxmlsec1.so.1
#4  0x00007fcdcb42a41dc4 in PyXmlSec_TreeAddIds (self=<optimized out>, args=<optimized out>, kwargs=<optimized out>)
    at /root/.cache/uv/sdists-v8/pypi/xmlsec/1.3.13/yz7dV7XYQgBxhXxL3Xsjk/src/src/tree.c:192
#5  0x000056420197cb73 in cfunction_call (func=0x7fcdcb429da0, args=<optimized out>, kwargs=<optimized out>) at Objects/methodobject.c:100
#6  0x000056420191eb40 in _PyObject_MakeTpCall (tstate=0x564201e84610 <_PyRuntime+458992>, callable=callable@entry=0x7fcdcb429da0,
    keywords=0x0) at Objects/call.c:240
#7  0x000056420191f496 in _PyObject_VectorcallTstate (kwnames=<optimized out>, nargsf=<optimized out>, args=0x7fcdcb435a0a8, callable=callable@entry=0x7fcdcb429da0)
    at ./Include/internal/pycore_call.h:90
#8  0x00005642018b8fac in _PyEval_EvalFrameDefault (tstate=<optimized out>, frame=0x7fcdcb4359ff8, throwflag=<optimized out>) at Python/ceval.c:177
#9  0x0000564201923678 in _PyObject_VectorcallTstate (kwnames=0x7fcdcb4359ff8, nargsf=2, args=0x7fcdcb4359ff8, callable=0x7fcdcb4359ff8)
    at ./Include/internal/pycore_call.h:92
#10 method_vectorcall (method=<optimized out>, args=0x7fcdcb4359ff8, nargsf=<optimized out>, kwnames=0x7fcdcb4359ff8) at Objects/classobject.c:100
#11 0x000056420192175b in PyObject_Call () at Objects/call.c:388
#12 0x00005642018b45d7 in _PyEval_EvalFrameDefault (tstate=<optimized out>, frame=0x7fcdcb4359b60, throwflag=<optimized out>) at Python/ceval.c:177
#13 0x0000564201923678 in _PyObject_VectorcallTstate (kwnames=0x7fcdcb4359b60, nargsf=2, args=0x7fcdcb4359b60, callable=0x7fcdcb4359b60)
```

python-xmlsec



Python bindings for the [XML Security Library](#).

We use this package as part of with SAML signature validation process

```
[[package]]
name = "python3-saml"
version = "1.16.0"
source = { registry = "https://pypi.org/simple" }
dependencies = [
    { name = "isodate" },
    { name = "lxml" },
    { name = "xmlsec" },
]
```

```
56     class OneLogin_Saml2_Utils(object):
790         def validate_sign(xml, cert=None, fingerprint=None, fingerprintalg='sha1', validatecert=False, debug=False):
814
815             :param multicerts: Multiple public certs
816             :type: list
817
818             :param raise_exceptions: Whether to return false on failure or raise an exception
819             :type raise_exceptions: Boolean
820             """
821             if xml is None or xml == '':
822                 raise Exception('Empty string supplied as input')
823
824             elem = OneLogin_Saml2_XML.to_etree(xml)
825             xmlsec.enable_debug_trace(debug)
826             xmlsec.tree.add_ids(elem, ids: ["ID"])
827
```

lxml 5.0.0 incompatibility - Arithmetic exception #277

✓ Closed



eljeffeg (ElJefte) opened on Jan 4, 2024



[lxml](#) 5.0.0 was released on Dec 29th 2023. This update causes an Arithmetic exception (core dumped) when using python-xmlsec.

Example code that would fail..

```
import xmlsec
from lxml import etree
root = None
with open("sign4-in.xml") as f:
    root = etree.parse(f).getroot()
xmlsec.tree.add_ids(root, ["ID"])
```



👍 5



Assignees

No one assigned

Labels

No labels

Type

No type

Projects

No projects

Milestone

Minimal reproduce

```
import io
import xmlsec
from lxml import etree

stream = io.BytesIO(b'<?xml version="1.0" encoding="UTF-8"?>
<Envelope xmlns="urn:envelope" ID="ef115a20-cf73-11e5-aed1-3c15c2c2cc88">
  <Data>
    Hello, World!
  </Data>
</Envelope>
"')
root = etree.parse(stream).getroot()
xmlsec.tree.add_ids(root, ["ID"])
print(etree.tostring(root, pretty_print=True).decode())
```


We use lxml to create xml object and then sending this object to xmlsec

```
Python 3.12.9 (main, Mar 5 2025, 02:25:27) [GCC 11.4.0]
Type 'copyright', 'credits' or 'license' for more information
IPython 8.5.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: import io
...: import xmlsec
...: from lxml import etree
...:
...: stream = io.BytesIO(b"""<?xml version="1.0" encoding="UTF-8"?>
...: <Envelope xmlns="urn:envelope" ID="ef115a20-cf73-11e5-aed1-3c15c2c2cc88">
...:   <Data>
...: ^IHello, World!
...:   </Data>
...: </Envelope>
...: """)
...: root = etree.parse(stream).getroot()
...: xmlsec.tree.add_ids(root, ["ID"])
...: print(etree.tostring(root, pretty_print=True).decode())
Floating point exception (core dumped)
```

But why does this cause a crash?

Using different libxml2 versions together...

Block lxml binaries to avoid libxml2 incompatibilities #246

 Closed

xmlsec:master



nosnilmot:lxml-nobinaries



 Conversation 3

 Commits 1

 Checks 0

 Files changed 1



nosnilmot (Stu Tomlinson) on Dec 16, 2022

Contributor



When xmlsec and lxml are linked to different libxml2 versions there can be subtle compatibility issues. Forcing lxml to install from source causes it to link to same system version of libxml2 as xmlsec.

See bugs.launchpad.net/lxml/+bug/1960668 for the sorts of issues this should solve.

This problem occurs because xmlsec is linked to system libxml2 library, but lxml binaries bundle and link to a private copy of libxml2 that may be a different and incompatible version.



Reviewers – [review now](#)

Notifications

 None

3 participants



Required Dependency

You need to install `python3-saml`, this is included in the `saml` extra when installing `social-core`.

In case you run into `lxml & xmlsec libxml2 library version mismatch` error, it is caused by `lxml` being built against a different version of `libxml2` than `xmlsec`. To avoid this, please install both packages from the source and build them against system libraries:

```
# Install system dependencies  
sudo apt install libxmlsec1-dev  
  
# Install Python packages from the source  
pip install --no-binary lxml --no-binary xmlsec -e 'social-core[saml]'
```

Fix was to build lxml from source, so it will use the same system
libxml2 library

```
# This is done since lxml is using bundled libxml2 and not the one installed in system which cause issues with xmlsec  
# which use the system installed one. So we build lxml from source and not using the binary wheel file,  
# so it will use the same libxml2 as xmlsec package  
ENV UV_NO_BINARY_PACKAGE="lxml"
```

```

8 ▶ def test__lxml_is_using_the_system_libxml2():  ⚡ Yonatan Bitton
9     libxml2 = ctypes.CDLL("libxml2.so.2")
10    libxml2_version = ctypes.c_char_p.in_dll(libxml2, "xmlParserVersion").value.decode()
11    libxml2_version_tuple = int(libxml2_version[0]), int(libxml2_version[1:3]), int(libxml2_version[3:5])
12
13    assert libxml2_version_tuple == lxml.etree.LIBXML_VERSION == lxml.etree.LIBXML_COMPILED_VERSION
14
15
16 ▶ def test__xmlsec_not_crash_on_xml_parsing():  ⚡ Yonatan Bitton
17     stream = io.BytesIO(
18         b'<?xml version="1.0" encoding="UTF-8"?>
19         <Envelope xmlns="urn:envelope" ID="ef115a20-cf73-11e5-aed1-3c15c2c2cc88">
20             <Data>Hello, World!</Data>
21         </Envelope>
22         '
23     )
24     root = lxml.etree.parse(stream).getroot()
25
26     # This crash the python with `Floating point exception (core dumped)`
27     # in case of mismatch between libxml2 used by lxml and xmlsec
28     # Example in our case was:
29     #   libxml2           : 2.9.13
30     #   lxml version      : 5.3.1
31     #   lxml libxml used   : (2, 12, 9)
32     #   lxml libxml compiled : (2, 12, 9)
33     #   xmlsec version     : 1.3.13
34     xmlsec.tree.add_ids(root, ids=["ID"])

```

Stage 6: Production Rollout

Stage 6: Production Rollout

- QA approved, all checks passed 

Stage 6: Production Rollout

- QA approved, all checks passed 
- Begin with partial rollout (e.g., APM pods)

Stage 6: Production Rollout

- QA approved, all checks passed 
- Begin with partial rollout (e.g., APM pods)
- Gradual deployment by region — smallest to largest

Stage 6: Production Rollout

- QA approved, all checks passed 
- Begin with partial rollout (e.g., APM pods)
- Gradual deployment by region — smallest to largest
- Continuous monitoring:

Stage 6: Production Rollout

- QA approved, all checks passed 
- Begin with partial rollout (e.g., APM pods)
- Gradual deployment by region — smallest to largest
- Continuous monitoring:
 - No Sentry errors

Stage 6: Production Rollout

- QA approved, all checks passed 
- Begin with partial rollout (e.g., APM pods)
- Gradual deployment by region — smallest to largest
- Continuous monitoring:
 - No Sentry errors
 - No customer complaints

Stage 6: Production Rollout

- QA approved, all checks passed 
- Begin with partial rollout (e.g., APM pods)
- Gradual deployment by region — smallest to largest
- Continuous monitoring:
 - No Sentry errors
 - No customer complaints
 - System behaving as expected

Day 27

APM went well

Day 27

Australia too

Day 28

Even Europe...

Day 28

If all goes well until tomorrow, we will continue with US 🙌 ...

...But Things Didn't Go As Planned 🤖

New SSL Errors, customers complains they don't get their emails...



Email Delivery Failed

The emailing system could not deliver the email due to an error.

SSLError EOF occurred in violation of protocol

Rollback all production 🙄

Failing when we use STARTTLS, the context here is None

```
282         # non-secure connections.
283         if not self.use_ssl and self.use_tls:
284             self.connection.ehlo(name=settings.NAT_DNS_NAME)
285             context = None
286             if self.ssl_certfile and self.ssl_keyfile:
287                 context = ssl.create_default_context()
288                 context.load_cert_chain(certfile=self.ssl_certfile, keyfile=self.ssl_keyfile)
289             self.connection.starttls(context=context)
290             self.connection.ehlo(name=settings.NAT_DNS_NAME)
291         if self.username and self.password:
```

It's using the `stdlib` context method

```
769 self.ehlo_or_helo_if_needed()
770 if not self.has_extn("starttls"):
771     raise SMTPNotSupportedError(
772         "STARTTLS extension not supported by server.")
773 (resp, reply) = self.docmd("STARTTLS")
774 if resp == 220:
775     if not _have_ssl:
776         raise RuntimeError("No SSL support included in this Python")
777     if context is None:
778         context = ssl._create_stdlib_context()
779     self.sock = context.wrap_socket(self.sock,
```

Shouldn't it work, assuming the context is less restricted?

```
777  
778 # Backwards compatibility alias, even though it's not a public name.  
779 _create_stdlib_context = _create_unverified_context  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

"""Create a SSLContext object for Python stdlib modules

All Python stdlib modules shall use this function to create SSLContext objects in order to keep common settings in one place. The configuration is less restrict than create_default_context()'s to increase backward compatibility.

"""


```
682 * def create_default_context(purpose=Purpose.SERVER_AUTH, *, cafile=None,
683                               capath=None, cadata=None):
684     """Create a SSLContext object with default settings.
685
686     NOTE: The protocol and settings may change anytime without prior
687     deprecation. The values represent a fair balance between maximum
688     compatibility and security.
689     """
```

```
openssl s_client -connect fake.domain:25 -starttls smtp
```

```
-----
New, SSLv3, Cipher is DHE-RSA-AES256-SHA
Server public key is 4096 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
No ALPN negotiated
SSL-Session:
    Protocol  : TLSv1
    Cipher    : DHE-RSA-AES256-SHA
    Session-ID: 313B18B4F0944999D9F8F71354EA35210DF683DCD2B79FF977FBD8
    Session-ID-ctx:
    Master-Key: 8FB2E215E997197DFD32019200C2CBC9E0104545FACAB2A15E133A
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    Start Time: 1745393387
    Timeout    : 7200 (sec)
    Verify return code: 10 (certificate has expired)
    Extended master secret: no
-----
250 STARTTLS
```

```
server = smtplib.SMTP('fake.domain', local_hostname=settings.NAT_DNS_NAME)
server.set_debuglevel(1)
server.ehlo(name=settings.NAT_DNS_NAME)
context = ssl._create_stdlib_context()
context.minimum_version = ssl.TLSVersion.TLSv1
server.starttls(context=context)
server.quit()
```

```
1318         self.settimeout(None)
-> 1319         self._sslobj.do_handshake()
1320     finally:
1321         self.settimeout(timeout)
```

SSLError: [SSL: SSLV3_ALERT_HANDSHAKE_FAILURE] sslv3 alert handshake failure (_ssl.c:1010)

Maybe cipher related? looking for DHE-RSA-AES256-SHA

```
import ssl
context = ssl._create_stdlib_context()
print("\n".join(cipher['name'] for cipher in context.get_ciphers()))
```

Indeed not supported...

```
TLS_AES_256_GCM_SHA384
TLS_CHACHA20_POLY1305_SHA256
TLS_AES_128_GCM_SHA256
ECDHE-ECDSA-AES256-GCM-SHA384
ECDHE-RSA-AES256-GCM-SHA384
ECDHE-ECDSA-AES128-GCM-SHA256
ECDHE-RSA-AES128-GCM-SHA256
ECDHE-ECDSA-CHACHA20-POLY1305
ECDHE-RSA-CHACHA20-POLY1305
ECDHE-ECDSA-AES256-SHA384
ECDHE-RSA-AES256-SHA384
ECDHE-ECDSA-AES128-SHA256
ECDHE-RSA-AES128-SHA256
DHE-RSA-AES256-GCM-SHA384
DHE-RSA-AES128-GCM-SHA256
DHE-RSA-AES256-SHA256
DHE-RSA-AES128-SHA256
```

Weak TLS_DHE_RSA_WITH_AES_256_CBC_SHA

IANA name:

TLS_DHE_RSA_WITH_AES_256_CBC_SHA

OpenSSL name:

DHE-RSA-AES256-SHA

GnuTLS name:

- DHE-RSA-AES256-SHA uses **SHA-1**
- DHE-RSA-AES256-SHA256 uses **SHA-256**

But our installed openssl is supporting this type of cipher

```
In [24]: !openssl ciphers -v | grep DHE-RSA-AES256-SHA
ECDHE-RSA-AES256-SHA384 TLSv1.2 Kx=ECDH     Au=RSA  Enc=AES(256)  Mac=SHA384
DHE-RSA-AES256-SHA256  TLSv1.2 Kx=DH     Au=RSA  Enc=AES(256)  Mac=SHA256
ECDHE-RSA-AES256-SHA   TLSv1   Kx=ECDH     Au=RSA  Enc=AES(256)  Mac=SHA1
DHE-RSA-AES256-SHA     SSLv3   Kx=DH     Au=RSA  Enc=AES(256)  Mac=SHA1
```

Let's try again with missing cipher...

```
server = smtpplib.SMTP('fake.domain', local_hostname=settings.NAT_DNS_NAME)
server.set_debuglevel(1)
server.ehlo(name=settings.NAT_DNS_NAME)
context = ssl._create_stdlib_context()
context.minimum_version = ssl.TLSVersion.TLSv1
context.options |= ~ssl.OP_NO_SSLv3
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
context.set_ciphers(":".join([x["name"] for x in context.get_ciphers()] + [
    'DHE-RSA-AES256-SHA',
]))
server.starttls(context=context)
server.quit()
```



```
1319 self._sslobj.do_handshake()  
1320 finally:  
1321     self.settimeout(timeout)
```

```
SSL error: [SSL: DH_KEY_TOO_SMALL] dh key too small (_ssl.c:1010)
```

```
In [42]:
```

No client certificate CA names sent

Peer signing digest: MD5-SHA1

Peer signature type: RSA

Server Temp Key: DH, 1024 bits

SSL handshake has read 2715 bytes and written 552 bytes

Verification error: certificate has expired

New, SSLv3, Cipher is DHE-RSA-AES256-SHA

Server public key is 4096 bit

Secure Renegotiation IS supported

Compression: NONE

Expansion: NONE

No ALPN negotiated

SSL-Session:

Protocol : TLSv1

Cipher : DHE-RSA-AES256-SHA

Session-ID: E162D0531E5FF74FC7BC4CA14C2E0DD2D79789F73B4EDBF57D6

Session-ID-ctx:

But wait — we're already using OpenSSL v1, which is less strict by default. So why did these errors appear only in Python 3.12, but not in 3.9?

bpo-43998: Default to TLS 1.2 and increase cipher suite security (GH-25778) #25778

Merged

python:master



~~tiran:bpo-43762-tls12~~



on May 1, 2021



v3.10.0b1



Conversation 3

Commits 1

Checks 0

Files changed 8



tiran (Christian Heimes) on May 1, 2021 • edited

Member



The ssl module now has more secure default settings. Ciphers without forward secrecy or SHA-1 MAC are disabled by default. Security level 2 prohibits weak RSA, DH, and ECC keys with less than 112 bits of security.
:class: `~ssl.SSLContext` defaults to minimum protocol version TLS 1.2.
Settings are based on Hynek Schlawack's research.

Reviewers – [review now](#)



hynek

Notifications

None

All

✓ ↕ 43 ■■■■ Modules/_ssl.c

↑		@@ -152,15 +152,27 @@ extern const SSL_METHOD *TLSv1_2_method(void);
152	152	· · · · · #ifndef PY_SSL_DEFAULT_CIPHER_STRING
153	153	· · · · · #error "Py_SSL_DEFAULT_CIPHERS · 0 · needs · Py_SSL_DEFAULT_CIPHER_STRING"
154	154	· · · · · #endif
	155	· · · · · #ifndef PY_SSL_MIN_PROTOCOL
	156	· · · · · #define PY_SSL_MIN_PROTOCOL TLS1_2_VERSION
	157	· · · · · #endif

162	171	· */
163		· · #define PY_SSL_DEFAULT_CIPHER_STRING "DEFAULT:!aNULL:!eNULL:!MD5:!3DES:!DES:!RC4:!IDEA:!SEED:!aDSS:!SRP:!PSK"
172	+	· · #define PY_SSL_DEFAULT_CIPHER_STRING " <u>@SECLEVEL=2</u> :ECDH+AESGCM:ECDH+CHACHA20:ECDH+AES:DHE+AES:!aNULL:!eNULL:!aDSS:!SHA1:!AESCCM"
173		· · #ifndef PY_SSL_MIN_PROTOCOL
174		· · · · #define PY_SSL_MIN_PROTOCOL TLS1_2_VERSION

In Python 3.9, the default OpenSSL security level was 1. Starting from Python 3.10, the default is level 2 — even when using OpenSSL 1



SSL_CTX_set_security_level



🔍 Search

[Home](#) [FIPS-140](#) [Commands](#) [Libraries](#) [File Formats](#) [Overviews](#)

Libraries

[SSL_CTX_set_post_handshake...](#)
[SSL_CTX_set_psk_client_callba...](#)
[SSL_CTX_set_psk_find_session...](#)
[SSL_CTX_set_psk_server_callb...](#)
[SSL_CTX_set_psk_use_session...](#)
[SSL_CTX_set_quiet_shutdown](#)
[SSL_CTX_set_read_ahead](#)
[SSL_CTX_set_record_padding...](#)
[SSL_CTX_set_record_padding...](#)
[SSL_CTX_set_recv_max_early...](#)
[SSL_CTX_set_security_callback](#)

Security level set to 256 bits of security. As a result RSA, DSA and DH keys shorter than 15360 bits and ECC keys shorter than 512 bits are prohibited.

APPLICATION DEFINED SECURITY CALLBACKS

Documentation to be provided.

NOTES

The default security level can be configured when OpenSSL is compiled by setting -DOPENSSL_TLS_SECURITY_LEVEL=level. If not set then 1 is used.



SSL_CTX_set_security_level



[Home](#) [FIPS-140](#) [Commands](#) [Libraries](#) [File Formats](#) [Overviews](#)

Cipher suites not offering forward secrecy are prohibited. Session tickets are disabled.

Libraries

[SSL_CTX_set_post_handshake...](#)
[SSL_CTX_set_psk_client_callba...](#)
[SSL_CTX_set_psk_find_session...](#)
[SSL_CTX_set_psk_server_callb...](#)
[SSL_CTX_set_psk_use_session...](#)
[SSL_CTX_set_purpose](#)
[SSL_CTX_set_quiet_shutdown](#)
[SSL_CTX_set_read_ahead](#)
[SSL_CTX_set_record_padding_...](#)
[SSL_CTX_set_record_padding_...](#)
[SSL_CTX_set_recv_max_early_...](#)
[SSL_CTX_set_security_callback](#)
[SSL_CTX_set_security_level](#)
[SSL_CTX_set_session_cache_...](#)
[SSL_CTX_set_session_id_cont...](#)
[SSL_CTX_set_session_ticket_cb](#)
[SSL_CTX_set_split_send_fragm...](#)
[SSL_CTX_set_timeout](#)

• Level 4

Security level set to 192 bits of security. As a result RSA, DSA and DH keys shorter than 7680 bits and ECC keys shorter than 384 bits are prohibited. Cipher suites using SHA1 for the MAC are prohibited.

• Level 5

Security level set to 256 bits of security. As a result RSA, DSA and DH keys shorter than 15360 bits and ECC keys shorter than 512 bits are prohibited.

APPLICATION DEFINED SECURITY CALLBACKS

Documentation to be provided.

NOTES

The default security level can be configured when OpenSSL is compiled by setting -DOPENSSL_TLS_SECURITY_LEVEL=level. If not set then 2 is used.

So, for legacy SMTP servers, we explicitly allow weaker SSL connections
— but the default configuration is now much stricter

```
if self.use_weak_ssl_encryption:
    # Setting @SECLEVEL=1
    # Adding missing ciphers removed in python3.10
    # Support minimum TLSv1 version
    # etc.
    context = self.create_weak_ssl_context()
else:
    context = ssl.create_default_context()
```

Day 30

 **Python 3.12 fully deployed and stable in production.**

Takeaways

Stage 1 Takeaways: Planning & Environment Setup

- Plan before you start: prepare robust local environments for debugging
 - Audit your steps — documenting the path helps later troubleshooting
 - Prefer the official Python builds when possible; if compiling yourself, verify it thoroughly
-

Stage 2 Takeaways: Installing & Updating Packages

- Update to the latest package versions when compatible
 - Find the first version that supports Python 3.12 if needed
 - Some edge cases require manual overrides and patching
 - Be cautious with indirect dependencies — they may introduce silent conflicts
-

Stage 3 Takeaways: Runtime Failures & Debugging

- Many issues appear only at runtime — trial and error is inevitable
 - Compare behavior across versions to pinpoint changes
-

Stage 4 Takeaways: Testing in CI & Local

- Run your full test suite in **both** local and CI environments
 - If a test fails: don't patch blindly — it might hide a deeper problem
 - Treat differences seriously — don't ignore failing test
 - Even small Python upgrades may introduce meaningful changes in behavior
-

Stage 5 Takeaways: QA & Monitoring

- Use the full power of the QA team — they'll catch what tests missed
 - Expect bugs in edge cases and workflows you didn't simulate
 - Monitoring (logs, Sentry, metrics, customer feedback) is your safety net
-

Stage 6 Takeaways: Production Deployment

- Deploy slowly and incrementally — start with low-risk services
 - Production is always the **best testing environment** (for better or worse)
 - Stay alert and respond quickly — be ready with hotfixes as needed
-

Final Recap: Key Lessons from the Upgrade

- Upgrades are a journey, not a switch — plan, test, iterate
 - Python version changes can expose hidden tech debt
 - Good tooling (like uv, Sentry, CI) makes the process manageable
 - Stay curious, cautious, and collaborative
-

Thank You!
Any Questions?

bit.ly/py312-upgrade

Yonatan Bitton
@bityob

